

Examenul de bacalaureat național 2017
Proba E. d)
Informatică
Limbajul Pascal

Varianta 3

Filiera teoretică, profilul real, specializările: matematică-informatică

matematică-informatică intensiv informatică

Filiera vocațională, profilul militar, specializarea matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.

THEMA I

(30 Puncte)

Für Punkt 1, schreibt auf das Prüfungsblatt den Buchstaben welcher der richtigen Antwort entspricht.

1. Gebt den Pascal Ausdruck an, der den Wert `true` hat, dann und nur dann wenn die natürliche Zahl die in der ganzen Variablen `x` gespeichert wird aus genau drei Ziffern besteht. **(4P.)**

- a. `(x div 1000=0) and (x div 100<>0)` b. `(x div 1000=0) or (x div 100<>0)`
c. `(x mod 1000=0) and (x mod 100<>0)` d. `(x mod 1000=0) or (x mod 100<>0)`

2. Es sei der nebenstehende Pseudocode - Algorithmus.

Man beschriftet mit `a%b` den Rest der Teilung der natürlichen Zahl `a` durch die natürliche, von Null verschiedene Zahl `b`.

- a) Schreibt den angeschriebenen Wert, wenn die Zahlen 7, 4, in dieser Reihenfolge, gelesen werden. **(6P.)**

- b) Schreibt vier Eingabe – Datensets, natürliche Zahlen aus dem Intervall `[2,9]`, welche eingelesen werden können, so dass nach dem Durchlaufen des Algorithmus, für jeden Set der Wert 7 angezeigt wird. **(4P.)**

- c) Schreibt im Pseudocode einen, mit dem gegebenen äquivalenten Algorithmus, in dem ihr die `solange ...wiederhole` Struktur mit einer fußgesteuerten Wiederholungsstruktur ersetzt. **(6P.)**

```
lese a,b
    (natürliche Zahlen, a≥2, b≥2)
wenn a>b dann
| x←a
| a←b
| b←x
|
s←0
für x←a,b wiederhole
| c←2
| solange x%c>0 wiederhole
| | c←c+1
|
s←s+c
schreibe s
```

- d) Schreibt das dem gegebenen Algorithmus entsprechende Pascal Programm. **(10P.)**

THEMA II

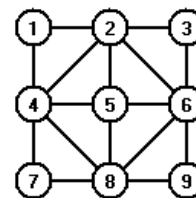
(30 Puncte)

Für jeden der Punkte 1 und 2 schreibt auf das Prüfungsblatt den Buchstaben welcher der richtigen Antwort entspricht.

1. Ein Baum mit 9 Spitzen, nummeriert von 1 bis 9, ist durch den „Vatervektor“ (5, 4, 6, 0, 3, 2, 6, 9, 7) dargestellt. Die Wurzel des Baums ist: **(4P.)**

a. 1 b. 4 c. 6 d. 8

2. In einem ungerichteten Graph sind zwei Kreise disjunkt wenn sie keine einzige gemeinsame Spitze haben. Für den nebenstehend dargestellten Graph mit 9 Spitzen, erzeugt man eine Menge gebildet aus elementaren Kreisen, mit der Eigenschaft dass jedwelche zwei von ihnen disjunkt sind. Die größte Anzahl von Kreisen aus so einer Menge ist: **(4P.)**



a. 1 b. 2 c. 3 d. 4

Schreibt auf das Prüfungsblatt die Antwort für jede der folgenden Anforderungen.

3. Die Variable **p** speichert gleichzeitig die Anzahl der Spitzen eines Polygons (natürliche Zahl aus dem Intervall $[3, 10^2]$) und die Koordinaten seiner Spitzen (Abszisse und Ordinate) im **xOy** Koordinatensystem (reelle Zahlen).

Wenn bekannt ist dass die untenstehenden Pascal Ausdrücke, als Werte die Anzahl der Spitzen eines Polygons haben, die Abszisse, beziehungsweise die Ordinate der ersten seiner Spitzen, schreibt die Definition eines Datentyps mit dem Namen **poligon**, ein Datensatz(Verbund) der das Speichern der Daten eines Polygons ermöglicht und deklariert entsprechend die Variable **p**.

```
p.numar
p.varf[1].x
p.varf[1].y
```

(6P.)

4. Die Variablen **i** und **j** sind vom Typ ganz, und die Variable **a** speichert ein zweidimensionales Feld mit 9 Zeilen und 9 Spalten, nummeriert von 1 bis 9, deren Elemente anfangs alle gleich mit dem Zeichen * sind.

Ohne andere Variablen zu verwenden, schreibt die untere Anweisungssequenz, indem die Auslassungspunkte so ersetzt werden, dass die Variable **a**, nach dem Durchlaufen der Sequenz, das nebenstehende Feld speichert.

```
for i:=1 to 9 do
  for j:=1 to 9 do
```

.....

(6P.)

```
b a a a a a a a a b
b b a a a a a b b
b b b a a a b b b
b b b b a b b b b
b b b b b b b b b
b b b b b b b b b
b b b b b b b b b
b b b b b b b b b
b b b b b b b b b
```

5. Gegeben ist ein Text mit höchstens 100 Zeichen, in dem die Wörter aus großen Buchstaben des englischen Alphabets bestehen und durch je eine Leerstelle getrennt sind. Schreibt ein Pascal Programm welches von der Tastatur einen Text vom oben beschriebenen Typ liest und auf den Bildschirm, in je eine Zeile, jene Wörter anschreibt, welche wenigstens eine Musiknote enthalten. Wenn es keine solche Wörter gibt, schreibt man auf den Bildschirm die Nachricht **nu exista**.

Die Musiknoten sind **DO, RE, MI, FA, SOL, LA, SI**.

Beispiel: für den Text

REMI DOMINO SI KHANHOO SUNT DENUMIRI DE JOCURI CE AU ASPECTE SIMILARE

werden die Wörter, nicht unbedingt in dieser Reihenfolge

REMI

DOMINO

SI

DENUMIRI

SIMILARE, angeschrieben.

(10P.)

THEMA III

(30 Puncte)

Für Punkt 1, schreibt auf das Prüfungsblatt den Buchstaben welcher der richtigen Antwort entspricht.

1. Unter Verwendung der Backtracking Methode erzeugt man, in steigender Reihenfolge, alle natürlichen Paarzahlen gebildet aus drei Ziffern, mit der Eigenschaft dass es keine zwei benachbarten, gleiche Ziffern gibt und die Summe der Ziffern 10 ist. Die ersten fünf erzeugten Zahlen, in dieser Reihenfolge sind: 136, 154, 172, 190, 208. Die sechste erzeugte Zahl ist: **(4P.)**

a. 217 b. 226 c. 262 d. 280

Schreibt auf das Prüfungsblatt die Antwort für jede der folgenden Anforderungen.

2. Das Unterprogramm **f** ist nebenstehend definiert. Schreibt was nach dem untenstehenden Aufruf angezeigt wird. **f(4);** **(6P.)**
- ```
procedure f(n:integer);
var i:integer;
begin
 if n>0 then begin
 for i:=1 to n do write(i);
 f(n-1)
 end
end;
```
3. Das Unterprogramm **identice** hat zwei Parameter, **a** und **b**, über die es je eine natürliche Zahl ( $10 \leq a \leq b \leq 10^6$ ) bekommt. Das Unterprogramm schreibt, auf den Bildschirm, alle natürlichen Zahlen aus dem Intervall **[a,b]** welche alle Ziffern identisch haben. Die angeschriebenen Zahlen sind durch je eine Leerstelle getrennt, und wenn es keine solche Zahlen gibt, schreibt man auf den Bildschirm die Nachricht **nu exista** an. Schreibt die vollständige Definition des Unterprogramms.  
**Exemplu:** für **a=700** und **b=1500** schreibt man auf den Bildschirm  
777 888 999 1111. **(10P.)**

4. Man nennt **Einfügen** einer Folge **A** in eine Folge **B** das Einfügen, zwischen zwei Glieder der Folge **B**, aller Elemente aus **A**, auf benachbarte Positionen, in der Reihenfolge in der sie in **A** erscheinen.

Die Datei **bac.in** enthält natürliche Zahlen aus dem Intervall **[1,10<sup>6</sup>]**: auf der ersten Zeile, die Zahlen **m** und **n**, und in jeder der folgenden zwei Zeilen, je eine Folge von **m**, beziehungsweise **n** ganzen Zahlen, in **streng steigender Reihenfolge** geordnet. Die Zahlen auf derselben Zeile der Datei sind durch je eine Leerstelle getrennt und die Nummerierung der Elemente in den Folgen beginnt von 1.

Man verlangt, das Anschreiben auf dem Bildschirm, der Lage aus der zweiten Folge, aus welcher die erste Folge eingefügt werden kann, so dass die erhaltene Folge streng steigend sein soll. Wenn es keine solche Lage gibt, wird auf dem Bildschirm die Nachricht **imposibil** angeschrieben. Verwende einen Algorithmus der in Bezug auf die Laufzeit und auf den benötigten Speicher effizient ist.

**Beispiel:** wenn die Datei die Zahlen

4 6

15 16 17 19

7 10 12 20 30 40

enthält, kann man die Folge 7, 10, 12, 15, 16, 17, 19, 20, 30, 40 erhalten und auf dem Bildschirm wird 4 angezeigt, und wenn die Datei, die Zahlen

4 6

15 16 17 19

7 14 18 20 30 40

oder die Zahlen

4 6

1 2 3 4

7 15 18 20 30 40

enthält, wird auf dem Bildschirm die Nachricht **imposibil** angezeigt.

a) Beschreibt in Umgangssprache den benötigten Algorithmus und begründet seine Effizienz. **(2P.)**

b) Schreibt das dem geschriebenen Algorithmus entsprechende Pascal Programm. **(8P.)**